|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **National University of Computer and Emerging Sciences, Lahore Campus** | | | | |
| final design | **Course:** | **Operating System Lab** | **Course Code:** | **CL205** |
| **Program:** | **BS(Computer Science)** | **Semester:** | **Fall 2018** |
| **Duration:** | **45 minutes** | **Total Marks:** | **15** |
| **Quiz Date:** | **26-Oct-2018** | **Weight** | **5** |
| **Section:** | **A** | **Page(s):** | **1** |
| **Exam:** | **Quiz 1** | **Roll No:** |  |

**Question # 1 (10)**

There are exactly 3 threads generate string a, b and c in an arbitrary order. In an absence of any synchronization mechanism there will be no order in generation of a, b and c. In the form of regular expression the string (a | b | c)\* {\* means many times a character an occur, | means or, so different combinations can be aaaaaaa… , bbbbbbbbbbb… , ccccc… }. Synchronize threads using semaphore in such a way that your printed string will be (cbbbba)\* {\* means many times cbbbba can occur, so different combinations will be cbbbba, cbbbba , cbbbba,….}.

Note you are not allowed to add or delete any cout statement

|  |  |  |
| --- | --- | --- |
| //thread 1  While(1)  {    Cout << ‘a’;  } | //thread 2  While(1)  {  Cout << ‘b’;  } | //thread 3  While(1)  {  Cout << ‘c’;  } |